Pengantar Sain Data

Konsep dan Aplikasi untuk Bisnis

# Clustering

# 2. Fuzzy C-Means Clustering (FCM)

### 2.1 Apa itu FCM?

**Fuzzy C-Means (FCM)** adalah metode *clustering* atau pengelompokan data yang berbasis pada logika fuzzy. Berbeda dengan metode klusterisasi klasik seperti **K-Means**, di mana satu data hanya bisa menjadi anggota **satu kluster saja**, dalam FCM, sebuah data dapat menjadi anggota dari **lebih dari satu kluster** dengan **derajat keanggotaan tertentu**.

Tujuan utama FCM: Mencari pusat kluster sedemikian rupa sehingga setiap data memiliki derajat keanggotaan terhadap tiap kluster.

### 2.2 Karakteristik FCM

| Fitur | Deskripsi |
| --- | --- |
| **Metode Klusterisasi** | Unsupervised learning (tidak memerlukan label) |
| **Jenis Klusterisasi** | Fuzzy (soft clustering) |
| **Derajat Keanggotaan** | Data bisa termasuk ke lebih dari satu kluster |
| **Output** | Matriks derajat keanggotaan + pusat kluster |

### 2.3 Prinsip Kerja FCM

FCM bekerja dengan prinsip **optimisasi fungsi tujuan (objective function)** berikut:

dimana:

- : titik data ke-

- : pusat kluster ke-

- : derajat keanggotaan pada kluster

- : eksponen fuzziness (), biasanya

- : jumlah kluster - : jumlah data

Tujuan: Minimalkan nilai fungsi → cari nilai dan yang optimal.

### 2.4 Algoritma FCM

1. **Inisialisasi**
   * Pilih jumlah kluster
   * Pilih nilai fuzzifier
   * Inisialisasi matriks keanggotaan awal secara acak, dengan syarat:
2. **Hitung pusat kluster**
3. **Update derajat keanggotaan**
4. **Cek konvergensi**
   * Hitung selisih antara matriks keanggotaan sebelum dan sesudah update.
   * Jika perubahan < toleransi (), hentikan proses.
5. **Iterasi ulang**
   * Kembali ke langkah 2 sampai algoritma konvergen.
6. **Tentukan kluster akhir**
   * Untuk setiap data , tentukan kluster utamanya berdasarkan nilai tertinggi.

### 2.5 Kelebihan FCM

* Lebih fleksibel daripada K-Means karena mengizinkan keanggotaan parsial.
* Cocok untuk data yang tidak jelas batasnya antar kluster.
* Memberikan informasi ketidakpastian tentang keanggotaan kluster.

### 2.6 Kekurangan FCM

* Sensitif terhadap inisialisasi awal.
* Perlu menentukan jumlah kluster secara manual.
* Komputasi agak lambat untuk dataset besar.
* Hasil bisa berbeda-beda setiap kali running karena inisialisasi acak.

### 2.7 Contoh Aplikasi FCM

| Bidang | Contoh Penggunaan |
| --- | --- |
| **Medis** | Mengelompokkan pasien berdasarkan gejala |
| **Pemasaran** | Segmentasi pelanggan |
| **Image Processing** | Segmentasi warna/gambar |
| **Keuangan** | Analisis risiko investasi yang bersifat tidak pasti |
| **Data Mining** | Menemukan pola dalam data yang tidak jelas strukturnya |

### 2.8 Visualisasi Intuitif

Bayangkan kamu punya kumpulan titik data di bidang 2D. Dengan FCM, kamu bisa membuat beberapa “pusat tarikan” (kluster), dan setiap titik ditarik oleh semua pusat ini, tapi dengan tingkat tarikan (derajat keanggotaan) yang berbeda-beda.

### 2.9 Implementasi (Python)

from fcmeans import FCM  
import numpy as np  
  
# Data contoh  
X = np.array([[1, 2], [2, 3], [3, 4], [8, 9], [9, 10]])  
  
# Buat model FCM  
fcm = FCM(n\_clusters=2)  
fcm.fit(X)  
  
# Hasil  
centers = fcm.centers  
labels = fcm.predict(X)  
membership = fcm.u # Matriks derajat keanggotaan

### 2.10 Referensi

* Bezdek, J.C., Ehrlich, R., & Full, W. (1984). **FCM: The Fuzzy C-Means Clustering Algorithm**. Pattern Recognition Letters.

**Contoh**

#### Dataset:

X = [  
 [1, 2],  
 [2, 3],  
 [3, 4],  
 [8, 9],  
 [9, 10]  
]

* Jumlah data:
* Dimensi fitur: 2
* Jumlah kluster:
* Eksponen fuzziness:
* Toleransi konvergensi:

#### Langkah 1: Inisialisasi Matriks Derajat Keanggotaan Awal

Baris 1 = derajat keanggotaan terhadap kluster 1  
Baris 2 = derajat keanggotaan terhadap kluster 2

#### Langkah 2: Hitung Pusat Kluster Awal

Gunakan rumus:

#### Untuk kluster 1:

#### Pembilang:

* :
* :

#### Penyebut:

Jadi:

#### Untuk kluster 2:

#### Pembilang:

* :
* :

#### Penyebut:

Jadi:

Pusat kluster awal:

#### Langkah 3: Update Matriks Derajat Keanggotaan

Gunakan rumus:

Karena , pangkatnya menjadi 2.

Contoh: Hitung , yaitu derajat keanggotaan titik pertama pada kluster 1.

#### Jarak dari ke pusat kluster:

#### Hitung :

Dan:

Lanjutkan perhitungan untuk semua data dan kluster.

#### Hasil Matriks Keanggotaan Baru (Setelah Iterasi 1)

Misalkan setelah dihitung semua:

#### Langkah 4: Hitung Ulang Pusat Kluster

Gunakan untuk menghitung ulang pusat kluster.

#### Untuk kluster 1:

Hitung pembilang dan penyebut seperti sebelumnya:

* :
* :

Penyebut:

Jadi:

#### Untuk kluster 2:

Lakukan hal yang sama, hasil akhir:

#### Langkah 5: Iterasi Sampai Konvergen

Ulangi langkah-langkah: 1. Gunakan dan baru untuk hitung jarak. 2. Hitung ulang . 3. Bandingkan selisih antara dan . Jika < , hentikan.

#### Hasil Akhir Setelah Beberapa Iterasi (Contoh)

Setelah beberapa iterasi, algoritma konvergen:

#### Pusat Kluster Akhir:

#### Matriks Derajat Keanggotaan Akhir:

#### Label Kluster Utama:

Artinya: - Titik 1–3 → kluster 0 - Titik 4–5 → kluster 1

#### Implementasi Python (Contoh 5 data dua fitur

from fcmeans import FCM  
import numpy as np  
# Data  
X = np.array([[1, 2], [2, 3], [3, 4], [8, 9], [9, 10]])  
# Model FCM  
fcm = FCM(n\_clusters=2)  
fcm.fit(X)  
# Hasil  
centers = fcm.centers  
labels = fcm.predict(X)  
membership = fcm.u  
print("Pusat kluster:", centers)  
print("Label kluster:", labels)  
print("Derajat keanggotaan:\n", membership)

Output contoh:

Pusat kluster: [[2.0 3.0]  
 [8.5 9.5]]  
Label kluster: [0 0 0 1 1]  
Derajat keanggotaan:  
 [[0.999 0.997 0.990 0.010 0.003]  
 [0.001 0.003 0.010 0.990 0.997]]

#### Kesimpulan

| Titik | Koordinat | Kluster Utama | Derajat Keanggotaan Kluster 1 | Derajat Keanggotaan Kluster 2 |
| --- | --- | --- | --- | --- |
| 1 | (1,2) | 0 | 0.999 | 0.001 |
| 2 | (2,3) | 0 | 0.997 | 0.003 |
| 3 | (3,4) | 0 | 0.990 | 0.010 |
| 4 | (8,9) | 1 | 0.010 | 0.990 |
| 5 | (9,10) | 1 | 0.003 | 0.997 |